MAFIA: MALICIOUS FACEBOOK PAGE IDENTIFICATION

A Thesis

by

VISVANATHAN THOTHATHRI

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirement for the degree of

MASTER OF SCIENCE

Chair of Committee,	Guofei Gu
Committee Members,	A.L.Narasimha Reddy
	Yoonsuck Choe
Head of Department,	Dilma Da Silva

December 2016

Major Subject: Computer Engineering

ABSTRACT

Facebook has been a host to many different attack vectors, such as malicious apps that use their permissions to spread malware and compromise user accounts to spam inboxes and walls. We have discovered another method of launching attacks that is sneakier and less intrusive - Facebook pages. Facebook pages largely draw users by promising interesting features or downloads; they do not require their audience to provide anything in return, and they use their large followings to distribute and promote malware. Facebook pages are easy to create, are not verified for safety/authenticity, and need no permissions. They vastly outnumber Facebook applications, and collude in a less obvious way.

This work attempts to establish the legitimate nature of this attack channel as a cause for concern and determine a method to expose the vectors. We propose **MAFIA** - *Malicious Facebook Page Identification*, which uses machine learning and careful system analysis to determine which Facebook pages are likely to distribute malware. Due to the nature of their deception, we find some mention worthy features and trends, and see that a significant number of users are exposed to these threats on a daily basis.

We propose using the Page-friend graphs to gain further insight into the nature of Page relationships shown by benign vs malicious pages. Since Facebook pages have a different set of restrictions (unlike Twitter or LinkedIn) regarding friend-relationships, we utilize the reciprocal-edge graph between pages and their posting information with decision-tree based variants to accurately determine what features contribute the most to

the identification of malicious pages in the Facebook ecosystem. Initial implementation and results reflect on the efficacy of our system.

This work is dedicated to my mother, father, little brother, and my best friends - for their support, love, and encouragement. For helping me remember what I'd forgotten.

ACKNOWLEDGMENTS

My master's thesis has been a long and enriching journey, and I have learnt a lot from many of my professors, peers and friends.

Primary gratitude goes to my advisor, Dr. Guofei Gu, for giving me this opportunity to step into a new field and really understand what research is all about. By expecting a high standard from me, he gave me cause to expect myself to always perform to a high standard. His approach helped me strip away the unnecessary frills of any idea and critically analyze any problem - and leaves me with much faith in myself to approach problems from different angles.

Secondly, I would like to thank Dr. Narasimha Reddy and Dr. Yoonsuck Choe for taking an interest in my research, and letting me establish my work on solid ground. I have to extend my thanks to my mentor, Jialong Zhang, for all his help in bouncing ideas with me and discussing the finer points of my project. The other members of the SUCCESS lab I'd like to thank - Srinath Nadimpalli and Xu Yan for lending me their ears and advice from time to time, Abner, Patrick and Lei for their encouragement, and David for being a lab mate who made often-tedious research lab hours much more tolerable. Many thanks must be extended to my friend Krishna who introduced me to the wonderful world of Golang.

My Uncles Ram and Sridhar, who gave me a lot of hope during times when I couldn't find a lot on my own.

I would also like to take this opportunity to thank my parents, brother and grandfathers, and best friends for their undying faith.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

What started with classmates.com, AOL and Friendster has now become the sensation we know as the online social network, and it's current vanguard is Facebook, Twitter, and LinkedIn. These have evolved from purely informational tools to know the whereabouts of former friends and acquaintances, to all-encompassing platforms that are used for entertainment, event organization, discussion, and information dissemination.

In 2006, Symantec produced a report with hints that hackers were turning to online social networks as a viable medium for distributing malware. Jagatic et. al harvested publicly-available information from Facebook and used it for email-based attacks. It was found that pretending to be a friend of the sender made phishing attacks significantly more successful in terms of click-through rate. In 2007, Brown et. al did a study of context-aware email spam based on profile information that is unrestricted to the general public, using automated html parsing.

Vern Paxson et. al identified Facebook and Twitter spam using a series of honeypots in 2010[6]. They identified several types of spammers based on posting frequency, how they selected the people on their list, and message similarity features. This work was before facebook had detailed categories of data visibility based on the level of closeness of the observer (acquaintance, friend, family, etc.). The Koobface botnet largely relied on social networks to spread its binaries, using a different one for each site and mainly using social engineering.

Thus, in the larger scale of OSNs, many channels of attack have presented themselves, and we find ourselves with a unique dilemma for Facebook. Users have an account that represents their identity or footprint on these website, and a page to themselves where they can present information about themselves. The level of information they choose to share with friends, the general public, and any applications they choose to use is determined by the user, in the form of permissions and access lists. Several years ago, when applications used to be more prominent, Facebook permissions were a lot less granular, and allowed accesss to the user's personal inbox, friend list, posting on their wall, etc. An earlier study that exposed the presence of a large number of applications using their permissions to generate spam and compromise user security resulted in a crackdown from Facebook. These have made it significantly more difficult to generate any applications with ill intent, since Facebook users now have the right to refuse any requested permission, and a denied permission is only repealed by Facebook after manual inspection by multiple staff members.

While this has significantly stifled malicious applications, facebook pages on the other hand continue to run amok because they operate completely independent of this system. Pages are followed by interested users, and require no permissions - they become popular mainly because the subject on which the page is based is of interest to a particular fan base. There are 6 main categories for pages (Public Figure, Place, Brand, Entertainment, Community, Organization) each of which has around 25 sub categories. Users *Like* a page to show their interest in its content, and promote it of their own free will. Unlike

user friendlists(which are restricted to 5000 friends), a fan page can have any number of people who subscribe to its content. Combined with the fact that people often share posts by these pages of their own volition, we have a potentially dangerous mixture that could easily see a large spread of malware.

In this work, we attempt to address the problem of pages spreading malicious data by identifying a unique feature set. We then present a comparison with existing works to show that our feature set is superior to past research and leverage graph features to identify trends in malicious pages and their neighbor networks.

The rest of the thesis is organized as follows - Chapter 2 contains a thorough explanation of the ecosystem followed by an in-depth analysis of the related works and background information in Chapter 3. We present the design and implementation of our system in Chapter 4, and evaluate our results in Chapter 5. Limitations and future work are recognized in Chapter 6.

2 BACKGROUND INFORMATION

2.1 Facebook Ecosystem - A Primer

In this section, we provide an introduction to the Facebook OSN to provide context for the reader regarding the rest of the thesis.

Users, Feeds, and Walls

Logging into Facebook.com we see that each user has a *news feed* where information pops up - this is the default home page and shows information from pages followed, apps liked, users followed or friended, and groups or communities they are a member of.

Buttons on the top bar allow users to easily shift between their homescreens. One such screen shows the user's *wall*, which is where personalized posts can be made by friends. Formerly, there was a permission that allowed facebook applications to post on user's walls but this was removed after an overhaul in permission granularity with an upgrade in facebook's graph API.

Facebook Graph API

The Graph API is the primary way of submitting and retrieving data from the facebook platform. It is an HTTP API that can be used to query for data, submit stories, and a variety of other tasks - with the help of an access token and an unique number representing the app/page/user, called their ID. These IDs are usually between 9 and 16 digits long and are assigned at random(or at least in a pseudo random manner unknown to outside

research).

Facebook formerly used a RESTful API which is now deprecated, and has completely shifted to the Graph API, which is upgraded fairly often with major changes to the structure and access criteria. Most applications were built on v1.0 of the API and were lost once backward compatibility was removed for any apps that hadn't upgraded to at least v2.3 (as of this date). The latest version is now v2.7 and support for v2.(x) is provided up to two years from the release of v2.(x+1).

Versions of the API determine the structure of the 'tree' that determines access order for different edges and fields for a node of any kind (app/page/user). The availability of an edge is based on the presence of a user access token or app access token. User access tokens are generated based on a combination of permissions that the user is willing to grant regarding his/her profile and in turn allows them access to the public edges and fields of an app or page, such as recent posts, photos, videos, and status updates. An app access token typically is only available to the creator of the page/app or the head of a brand that involves all of these, and gives them avenue to page insights such as the number of recent visitors, the number of unique newsfeeds it reached, or the number of people who clicked on, reacted to, or shared a post from the page.

Creating a Facebook Page

The procedure involved in creating a facebook page is neither complicated nor time consuming. Steps involved are as follows:

1. From the main screen post-login, we click on the last drop down arrow and navigate

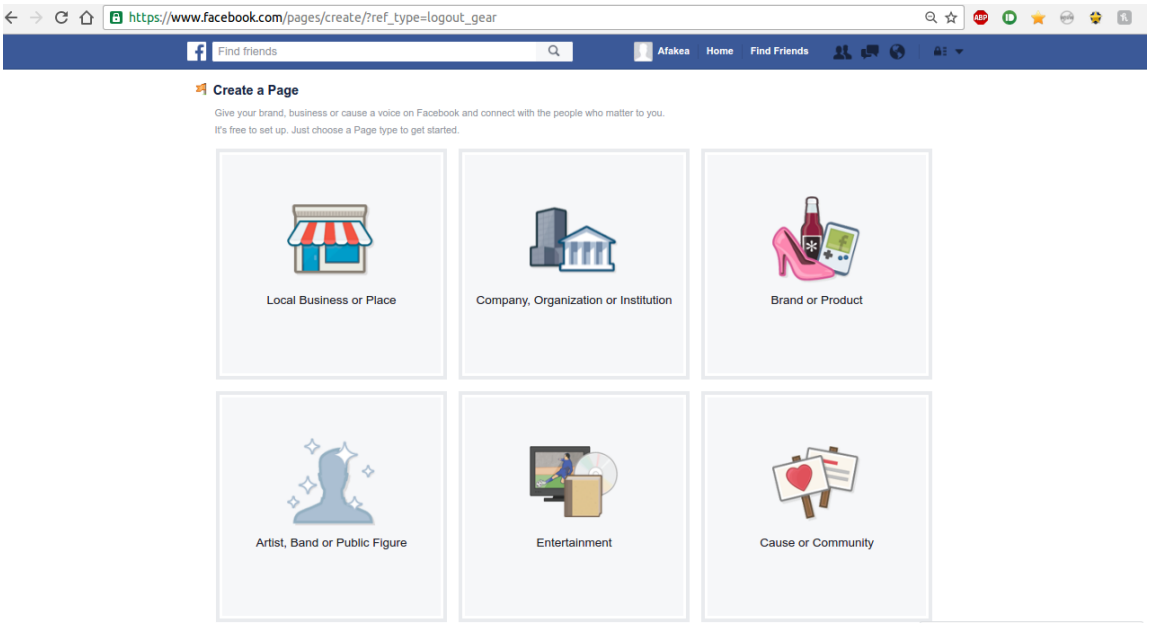


Figure 2.1: Making our Own Page - 1

to Create Pages. Here, we select which type of page it is on a broad spectrum (Place, Organization, Brand, etc.) out of 6 and then further, what category it falls into under each of these, shown in figure 2.1 and figure 2.2.

2. Each of these 6 has around 25 different categories based on what the page actually represents.

3. Now, we select a name, a profile picture, the target audience we begin advertising to, some basic details as well as the page url.

4. Our test page MAFIAPage is now set up (figure 2.3).

As is seen, we can now post content on the page. Another interesting feature is being

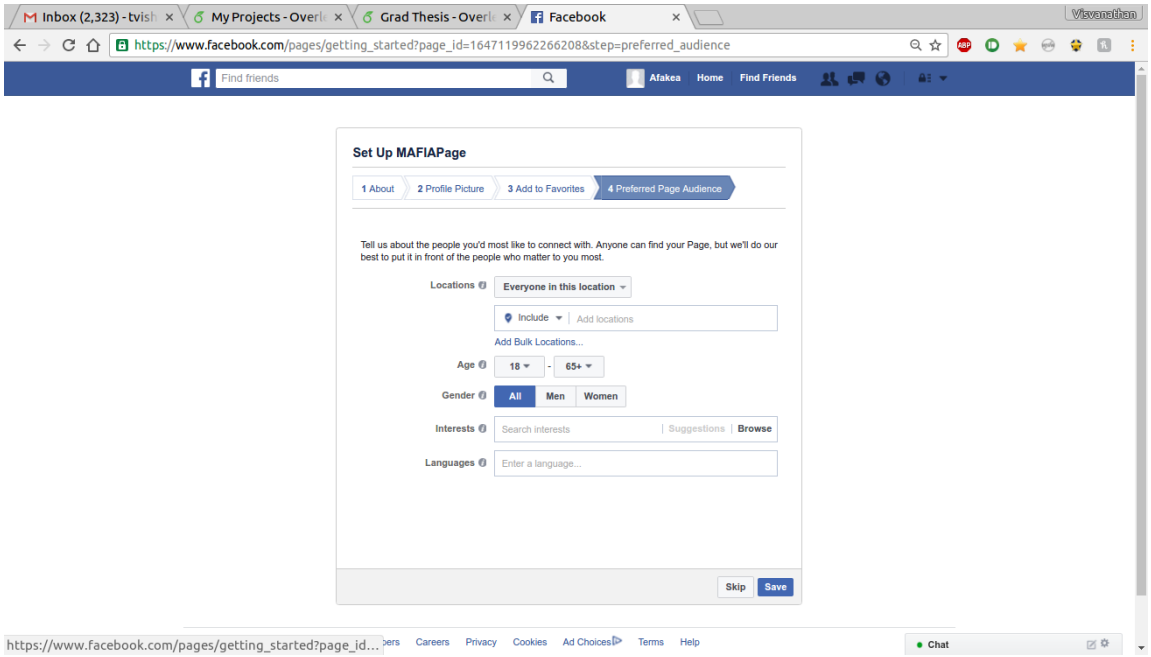


Figure 2.2: Making our Own Page - 2

able to like other pages as MAFIAPage, and have a page-newsfeed. After setting our preferred page audience, we can advertise it by means of artificial like inflation (covered in detail in the next section), or using the **Promote Menu** on the page as in figure 2.4.

Since Facebook advertises pages from the search bar based on a mixture of popularity and relevance. It further allows the user to specify which category they would like the results to be from (people, pages, photos, videos, groups, etc.). This is shown in figure 2.5.

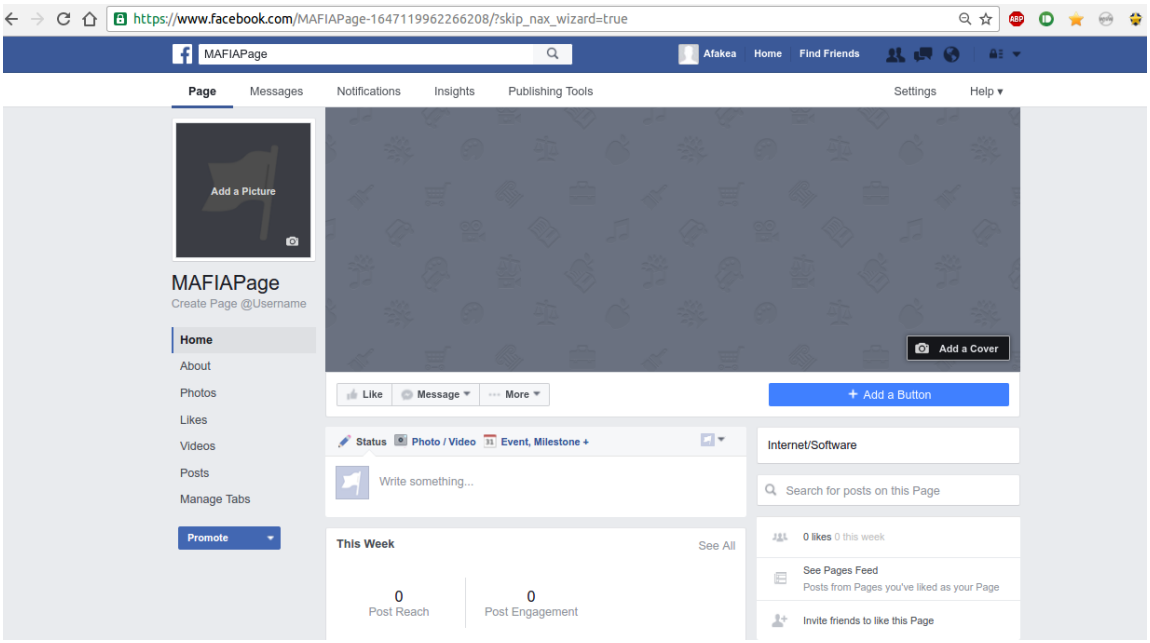


Figure 2.3: Completed Page!

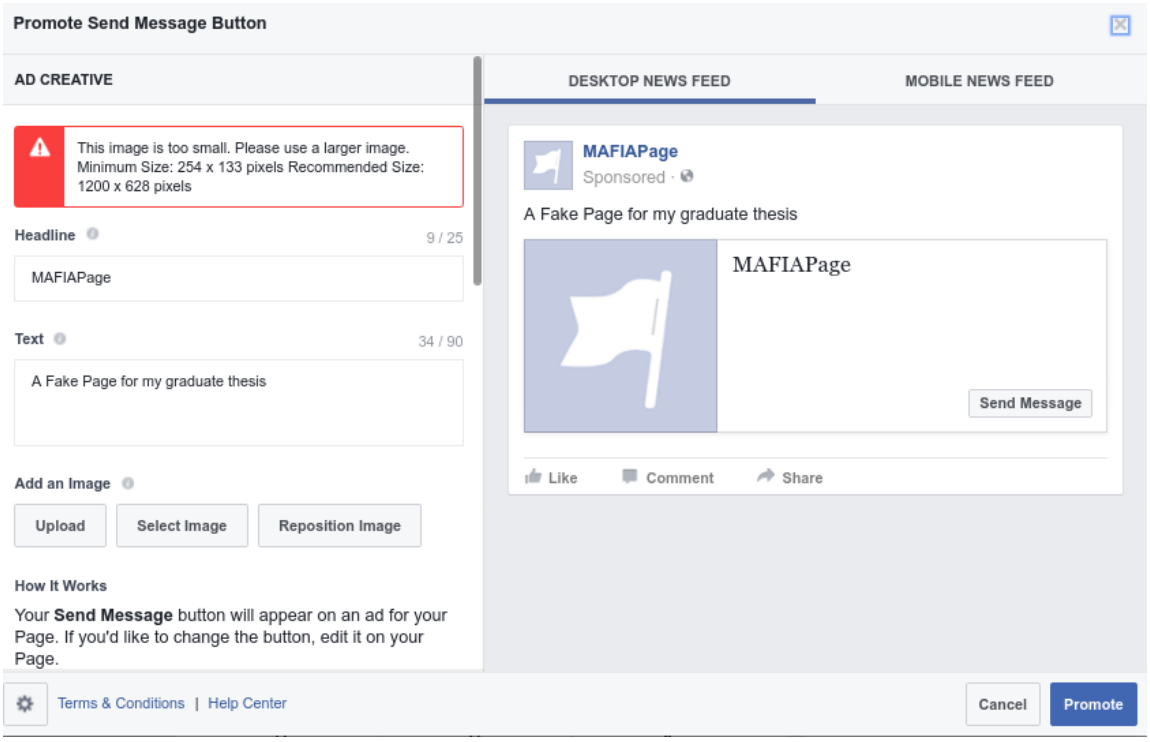


Figure 2.4: Promoting a Page

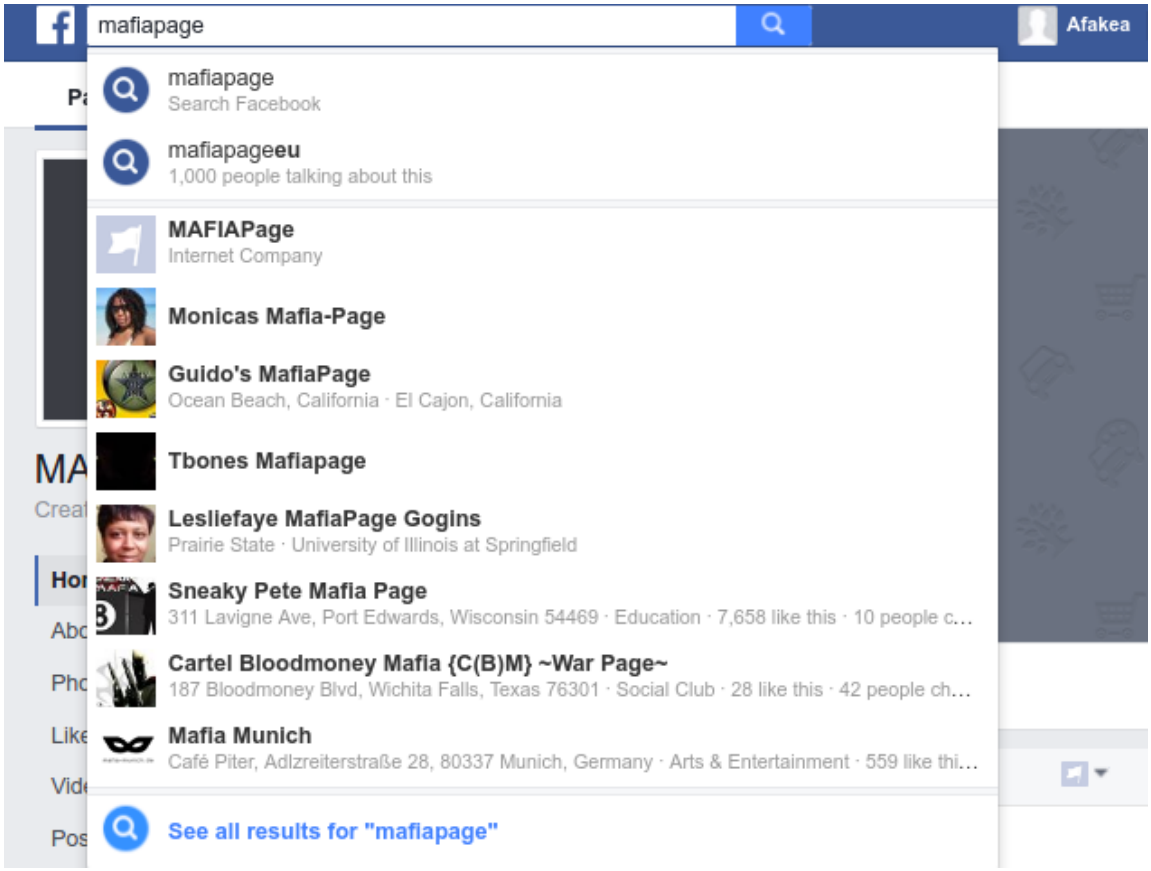


Figure 2.5: Search Results

3 RELATED WORK

Facebook has never been short of unscrupulous users who saw the platform as a means to spread malware in all its forms. From basic email spam, to taking advantage of natural disasters, there have been several works that specifically leverage the nature of Facebook's organization. Twitter, with its ease of user creation and simple follow-follower duality also suggests some insights into understanding the giveaways that expose malicious users.

In 2010, Facebook surpassed Google for daily user hits in the US. The same year, a hacker named Kirllos stole 1.5 million facebook accounts and offered to sell them underground at 2.5 cents each.

Facebook took it upon themselves to implement their own security measure in 2011, called the Facebook Immune System [11]. This studies the basic threat model on Facebook and implements a basic feature extraction system based on user behavior and common threats to the userspace and graph system. It was primarily machine learning based built on the idea that the facebook evasion model could be explained by adversarial learning and uses blacklists to filter urls out from entering the graph data stream.

Gao et. al from Northwestern University did a detailed study on the spread of spam campaigns based on these and studied 3.5 million users as well as 187 million wall posts [5]. Note that this study was before crawlers were required to get permission to engage with facebook. Their algorithm was based on identifying URL similarities (with and without

blacklists) and the similarity in meaning of the message outside of the URL. Based on these, they constructed user nodes that recommended the same URLs or had very similar messages to identify spam campaigns, by filtering for bursty posting and the distributed nature of users making these posts.

The nature of information distribution on facebook is such that distribution occurs in a tree-like fashion with malware cascading from one user to the next based on their friend lists, similar to the Twitter model [10] but without the anonymous-new user factor. Facebook promotes popular celebrities, but had no hashtag-based or popularity-based follow-suggestion as it seems to have adopted now.

The idea that facebook applications could be misusing their permissions and the data they were afforded therein was examined closely by Rahman et. al in their works on analyzing socware (social-network-malware) cascades [8] and Egele et. al's work PoX [4]. Socware is defined roughly as anything that tries to

1. compromise the user's device
2. asks for personal information
3. promises false rewards
4. asks for posting permissions that allow it to behave independently of the user's knowledge
5. points to a web page that requires the user to accomplish a task that profits the owner of the website
6. attempts to get the user to artificially inflate its reputation via free likes.

Egele et. Al proposed a clients-side proxy for privacy awareness which would also require the applications themselves to adopt these changes.

In [8] the authors developed an app called MyPageKeeper that focuses on all posters of malicious content based on contextual information whether they be apps or users or pages. This attempt focuses primarily on the idea that once authorized with a permission (or friendship) by a user, the offending party is free to post on the user's wall or (in some cases) sending private inbox messages to their contacts that are now endorsed by the user themselves. This piece of research prioritizes only the textual information on the post combined with the like information, across all users who allowed the mypagekeeper app. The interesting (and possibly disputable) idea with this paper is its classification of posts pointing to an app install, as well as posts asking for likes, as objectionable content, where they can also be viewed as purely advertising which while inconvenient, does not quite fit the billing of 'malware'.

Their follow up work FRAppE (Facebook Rigorous Application Evaluator) shifted its emphasis to the world of Facebook applications, which had roughly 20 million installations per day in 2012 [9]. It analyzes the posting behavior of 111k apps across 2.2 million facebook users and determines the collusion and propagation patterns of malicious applications. Creating an app is often as simple as using a $25 toolkit, and 13% of all the apps they analyzed were found to be malicious.

When a user adds an application, this process involves authorizing the application to view a portion of his profile information and giving it permission to perform certain

actions on his behalf - the granularity and scope of permissions offered has changed since then, but the idea is still true. This authorization is established in the form of an OAuth token, which allows the app to do so henceforth. MyPageKeeper only considers malware at post-level granularity, and 63% of malicious posts do not have an associated application - this leads into the work represented by FRAppE. Several of their features have a low level of robustness - such as profile description, permissions sought, etc. and could easily be circumvented by a more thorough attacker. Another area that could have been improved upon was the choice WoT (Web Of Trust) as an evaluator. WoT is highly community driven and without a rigorous method of evaluation is not a reliable base to declare benign/malicious. While Rahman et. al's results speak for themselves (93% accuracy), permissions have become far more granular and extremely stringent. Applications now have to prove that every permission they ask for is absolutely necessary, otherwise it is denied. Users can now decline to give certain permissions and apps must be capable of functioning around these as opposed to earlier (when the app would fail). New apps are also analyzed manually by Facebook developers before being allowed to hit the platform and open to all users.

In 2015, Dewan et. al [3] studied one of the most prevalent and ingenious attack methods on Facebook, which is taking advantage of its quick information dissemination which would otherwise be invaluable during disasters or major news events. Since people are often awaiting the latest updates on the social network of their choice, anyone pretending to be an authority could easily spread malware under the banner of seemingly legitimate

information. They studied 4.4 million posts in the wake of 17 major newsmaking events (such as disasters or terror attacks) and found that a majority of malicious content is from third party and web applications. This work focused on identifying malicious posts in real time with no informNation about them other than the content of the post and the accompanying metadata. While this approach seems to have been successful with an 86% detection rate it relies heavily on WoT, which is inherently biased.

Their follow-up work [2] tackles a similar problem to ours regarding the search for malicious pages but is different insofar as it has a much broader definition of malware, some of which are not traditionally consider malware but are disapproved of by Facebook such as hate speech, racial discrimination, political polarization, and misleading information. This work relies on users to judge whether these pages meet the above descriptions and WoT as a ground truth, which is unreliable and biased by users' personal beliefs.

This was also more of a measurement study which recognized that the origin of many malicious posts from their previous study was from a set of malicious pages, and attempted to identify these using purely n-gram bag of words detection.

De Cristofaro et al. undertook the task of identifying how pages and campaigns credence by artificial inflation of popularity measures such as likes [1]. Since more popular pages show up first and are promoted more heavily, like-farming as it is known becomes a significant threat to legitimate users attempting to popularize their campaigns. They found that there were two major types of modi operandi - bursty, bot based liking and stealthier behavior attempting to seem almost user-like in their liking behavior.

Viswanath et al. [12] studied the behaviors of sybil user accounts versus normal user accounts based on their spatio-temporal posting history and like activity. They determined that a large number of click-spam based likes were from anomalous users based on PCA subspace analysis.

In Twitter, two very influential works for ours were by Yang et al. on features that characterized Twitter spammers : [14] and [13]. These papers considered the neighbor-graph set up as well as the traditional tweet text-based features that have been seen in most previous works. While automation is not as significant on Facebook there were certainly a lot of inspirational ideas on these works.

Another related work on Twitter quickly realised that during times of urgency or crisis, people are much more likely to click links without much forethought, for instance in the wake of the tsunami in Japan [7].

4 DESIGN AND IMPLEMENTATION

To get our initial datasets, we start off with certain insights - easy ways to make users like a page that can potentially distribute malware to them are based on a *bait* - which we correctly guess to be one of the following: promotional facebook features (new colors/new button/new option), a superlative information comparison (who has the best ...?/ what is your most ...?), or download-esque clickbait. With these we manually verify our first set of malicious actors as well as benign (based on high popularity and beyond-reasonable-doubt) based on the content of any links/downloads/websites that are linked to. Figures 4.1 and 4.2 showcase some examples of pages distributing malicious content.

From here, we leverage the *Page Friends* edge to accrue a list of page IDs, and we use this procedure exponentially to get our entire dataset.

We now use the Koala Ruby API to get all the preprocessing data as json files by accessing the Facebook Graph Explorer. Using our custom code we calculate the feature values, which will be detailed shortly.

Using Virustotal's blacklists and Google SafeBrowsing as our ground truth, we examine all the links/downloadable files posted by all of these pages, and the links posted on all the first-level links. Our rationale here is that the page-creator is responsible for providing the link to a URL and is hence accountable for what content is found on it. We accumulate all these zeroth and first level links and test all these urls by feeding them into

our blacklists and testing for status.

Any page that is flagged once or more is marked as malicious, a metric that has been seen to work well for past related studies. We then use Gephi, a free data visualization software to consider the way the pages themselves are related and whether the page-graphs can give us any major insights. This provides our graph based features. Next, we combine these sets of features for each page in our benign and malicious data sets and run k-fold cross validation on our labeled data set.

4.1 FEATURES

4.1.1 Text-Based Features

Verified Or Not:

Facebook allows Pages which are locally or globally famous to be *verified* or accredited, which manifests itself as a tick next to the page name. The color of the tick is determined by the scope or reach of the brand - blue indicates international, black indicates local. This requires some form of verification by the Facebook authorities.

Average ASCII:

We measure the average number of non-ASCII characters across all posts from a page. We notice that a lot of pages empirically are seen to have several non-ASCII characters per post.

Maximum Post Similarity:

This is not a new feature, far from it - this is an obvious feature used in many works so far. We find it more useful to use this in tandem with average post similarity so that if the page tries to evade the conventional metric by reducing the number of post repeats to a small amount, the repeat is still caught.

Average Post Similarity:

A feature traditionally used in social networks - we use it in tandem with maximum post similarity to make better inferences. If avg similarity is low and max similarity is also low that likely indicates that a page is not trying to repeat the same message - hence not a low effort spammer.

Average Smileys:

Another interesting feature we notice is that a lot of these pages post a large number of smileys in their posts, with no discernible reason except to seem more amiable.

Average Unicode:

We use this to track the number of special unicode characters, which are observed to be more in number in malicious pages.

Stat Mean:

This represents the statistical mean of the time between posts. We calculate the time difference between every successive pair of posts and average these out, to see if the posts are predictable in this sense.

Stat Stddev:

Similar to the previous metric, this represents the statistial standard deviation of the time between successive posts, also looking for predictability through a low standard dev.

Fan Loyalty:

The idea behind this metric is to identify whether the page is largely like boosted or whether the fans are genuinely following the page. We first determine the average number of likes per post, and divide that by the total number of likes. Our intuition here is that if the pages are mainly campaign-promoted, there will be a comparatively small number of likes for each individual post - showing low levels of investment compared to the number of likes the page itself has.

Page Name Length:

In an earlier work it was discovered that a lot of malicious apps would typosquat on a more famous app's name which would make it easy to accidentally land on the bad app. Pages appear similarly in the drop down, but need to remain disparate from the legitimate page name for the same enterprise (for example, Cristiano Ronaldo Fan Page is likely official, so it is easier to add a suffix than to replace one letter and still remain seemingly above-board).

Non Small Letters:

Typing in many capitals or using many interesting characters catches the eye more readily than a properly syntactical and grammatical sentence. A lot of posts from malicious

pages do this since every click increases the likelihood that their malware spreads.

Number of Capital Letters:

In the same vein as the above, this metric counts attempts to 'bold' the post and convey an artificial sense of excitement.

AvgWhitespaceChars:

Many posts by malicious pages will often attempt to inflate post length compared to those by benign agents who are short and to the point.

Average Number of Links Per Post:

Since it makes sense intuitively to advertise more to get a product to spread faster, to get more clicks it would logically require more exposure - so malicious pages are often found to post urls in each post or post links to their websites exclusively.

Total Number of Links:

Related to the above, this is also an indication of how strong the promotion is. Neither Average nor Total on their own give a clear picture, so we calculate and include both.

Common Domain Count:

This is, as far as we know, unique to our work and is based on the idea that any campaign would primarily have an actor whose interests would be based off distributing malware from their own website/a few websites, as opposed to random links to malicious websites. We count the number of domains that hold at least 33% of all domains posted by that page in all its links/urls.

Most Common Story:

A previous work mentioned that malicious pages (though their definition of malicious was much looser) seemed to favor certain certain story types, such as ”Created Story” over others that were less easy to leverage since they did not let the page link to content of their own.

Most Common Post Type:

The idea behind this is similar to the last one - there will likely not be a lot of casual mobile status updates representing ’checkins’ or celebrating milestones - malicious pages are looking to hook users into activating their content.

Number of Pages Liked:

The intuition behind this metric is to determine whether a page is attempting to connect with relevant pages (from the same group, same field, etc.) or randomly maximizing the number of pages that it could gain followers from. We suspect this metric is useful as a correlative feature.

Likes from Other Pages:

We suspect that malicious communities are more willing to promote each other and hence would have actively liked or shared another page’s post.

People Talking About This:

People Talking About This (PTAT) is a metric introduced by facebook that keeps a note of how widespread a page’s reach was over a certain recent time period. It includes

the number of likes, stories generated by shares or views, number of newsfeed posts, etc. and is an interesting way of viewing a page's popularity.

***Likes and NewLikes*:**

These are a measure of the current total number of likes and the number of new distinct users who liked a page over the past week.

4.1.2 Graph-Based Features

Eccentricity:

Eccentricity is the maximum distance from a node to any other node it is connected to. In essence, the largest eccentricity value becomes the diameter of the graph. We expect that a graph with known malicious nodes will have differing values of eccentricity compared to the unknown ones.

Closeness Centrality:

Closeness centrality is a measure of how close any node on the social graph is to all other nodes on the graph, calculated as the sum of distances to all connected nodes.

Harmonic Closeness Centrality:

Harmonic Closeness Centrality is a variant of closeness centrality that accounts for the fact that a lot of network graphs might be sparse and hence largely incomplete. It uses the sum of the reciprocal of distances instead of the reciprocal of the sum, approximating the inverse of infinite to be 0. Since our graphs are expected to not be strongly connected, we expect there to be differences between the nature of page-friendships of a malicious

page vs a benign page.

Betweenness Centrality:

High betweenness centrality indicates that a page is a strong 'connector', or lies along the shortest path between many different pairs of nodes - essentially, the strongest point of contact.

Clustering Coefficient

Clustering coefficient is a measure of how connected a node's neighbors are as a fraction over the maximum possible connections that could each have. This determines how strongly a particular group of nodes is connected to each other and whether the nodes have several connections of their own or are united by connecting to a single node.

Eigencentrality:

Eigencentrality is similar to Pagerank. Starting all nodes with the same value (1/N), it percolates in a feedback loop where higher value nodes are good for connection and contribute more to the score of any particular node. Since many links are unidirectional we think this may offer some insight.

Reciprocal Indegree, Reciprocal Outdegree and Reciprocal Degree:

A big feature of our work is understanding how the reciprocal edge graphs contribute compared to standard edge comparison graphs. We notice a pattern between certain nodes that tend to act purely as 'reachers', liking pages that are popular without any reciprocation and use these features to quantify that instinct.

Indegree,Outdegree and Degree:

We contrast the above features with their counterpart values on the regular edge graph for a better understanding of the ecosystem.

Indegree Ratio and Outdegree Ratio:

Based on the ratio of the reciprocal edge-graph degrees to the degrees in the normal edge-graph.

Figure 4.1: A Bad Page

WhatsApp Hack - Account Spy Software

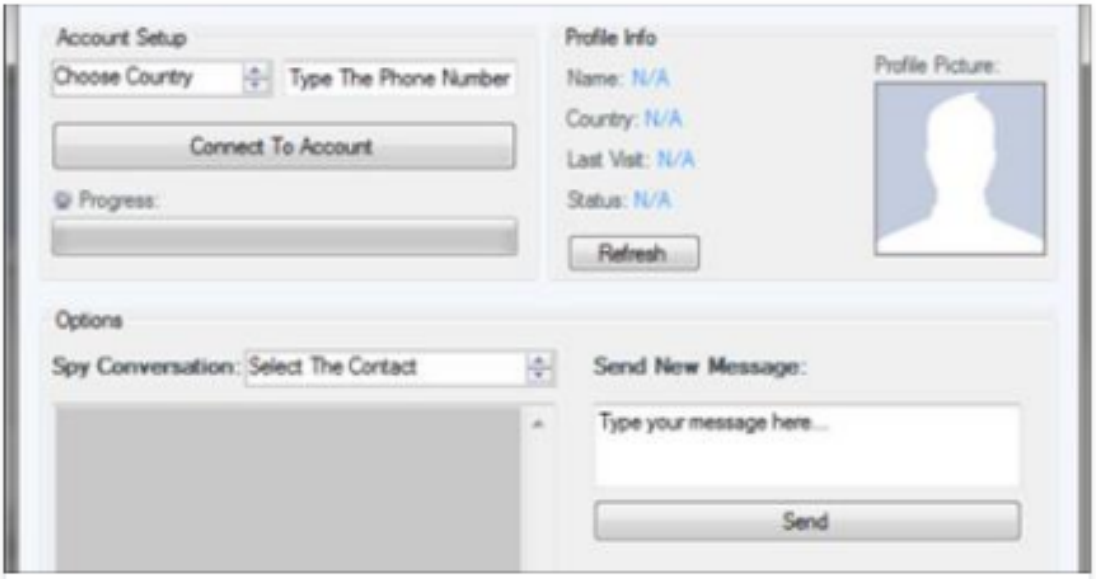
May 19, 2015 ·

We present you the WhatsApp Hack software created to spy over your desired contacts conversations and chats from WhatsApp messenger application. Developed by professional team of programmers who're authors of many similar apps from all over the web including some popular Android and iOS games, they decided to test their skills to create the ultimate software which can be used in spying purposes such as hacking WhatsApp accounts. After months of dedicated hard work they succeeded to make it work. Try out our Wapp spy program by visiting its official site from link below and have fun spying your friends chats!

> Download the App from official website:
http://wapphack.com/download/

Enjoy the tool!

—
#WhatsappHack #WhatsappAccountHack #WhatsappSpy



Download | WhatsApp Hack | Spy Tool

File Name: WAppSpy v.1.2.0 Setup.exeFile Size: 1MBSupported For:

Figure 4.2: Another Bad Page

5 EVALUATION AND RESULTS

5.1 Results And Insights

We use weka's algorithm base to evaluate our data using several different algorithms, tweaking hyperparameters as necessary. The area under the Receiver Operating Characteristics (ROC) curve, which plots the true positive rate as a function of the false positive rate, is used to determine the success of our work.

Using the randomforest algorithm, we get an AuC of 86% and an accuracy of 78% with a false positive rate of 11%. We find that adaptive boosting gives us up to 86% AuC with an f-measure is 0.76, and our best results are with a filtered classifier with an AuC of 87.1%, accuracy of 80.5% at an f-measure of 0.8. Detailed examination of our most effective features was done using several attribute selection methods and we found that betweenness centrality, reciprocal indegree, indegree, and several other graph features were quite prominent on the list.

These results are showcased in Figure 5.1 and Figure 5.2.

We also observed that most benign pages tend to act as 'reachers' - liking a middling number of pages and rarely being liked in return, primarily to show support (5-13).

Malicious pages primarily come in two forms - isolated pages and pages that function in group form, with multiple connections between them. The former kind functions independently and does not add much to our discussion of trends. The latter kind almost

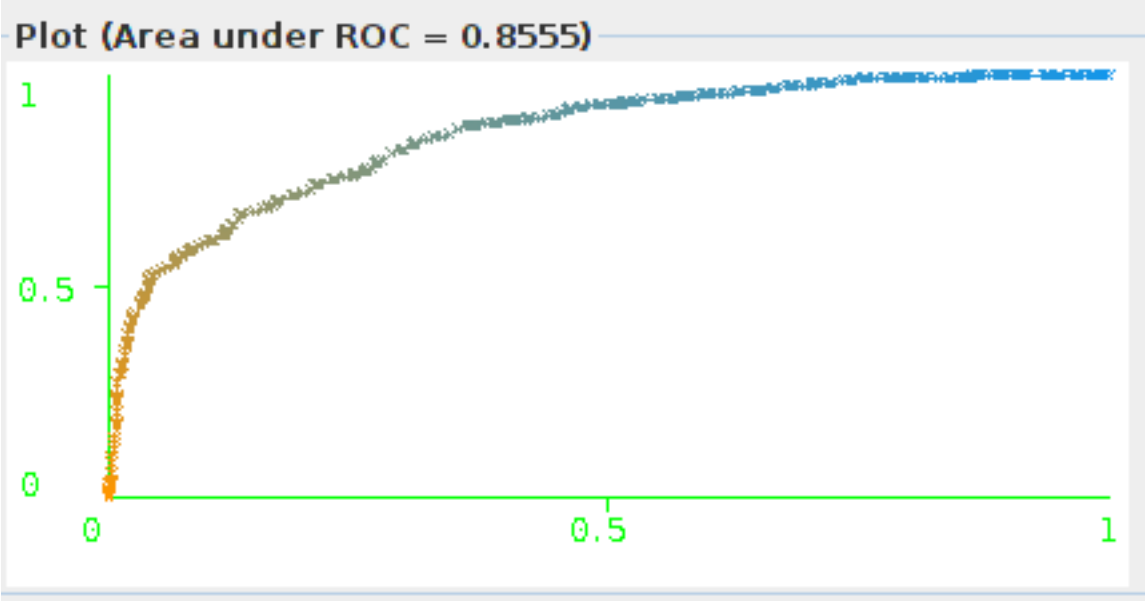


Figure 5.1: Results with Adaptive Boosting

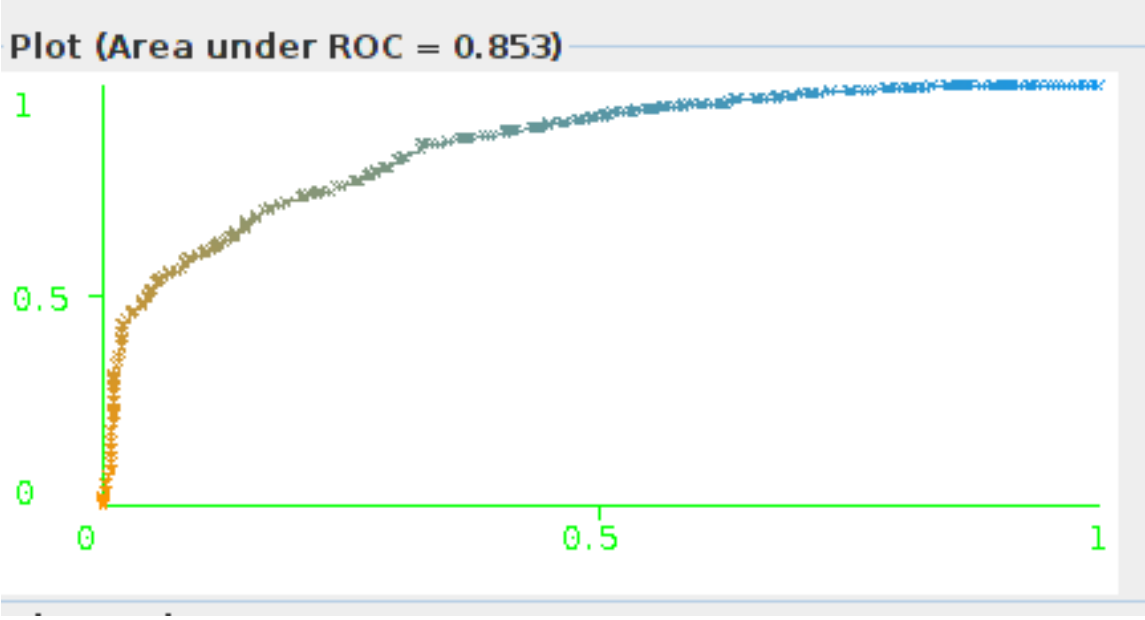


Figure 5.2: Results with Random Forest

always has 2 central nodes which are connected to several, while the others are all connected via these - this is verified by the reciprocal indegree and outdegree features which show some nodes with high betweenness centrality (central nodes). We also find that harmonic closeness centrality almost entirely consists of non zero values only in the malicious group since their networks are much better interconnected, this reduces the accessibility distances.

Most high-value benign nodes do not follow any pages - unless these pages are of the same verified group. They do not need to acquire followers from other pages since their provided value is high enough. These pages often have very high indegree and zero or very low outdegree.

Most benign nodes also have a very low eccentricity because of the way they are structured - mostly reachers with very few central nodes, most of which do not connect to other pages - which is closer to 0. Almost all non-zero eccentricities are for malicious nodes.

As far as common post types go, we find that previous trends are much less useful when dealing with actual malware - while links and photos are still the most common types, we observe an almost even split among benign and malicious nodes. The same goes for the common story types - added_photos and shared_story are the most common type, but along with mobile_status_update are almost even split - which suggests that a lot of inflammatory material (from politically charged entities or discriminatory pages in Dewan et. al's work) tends to skew towards not using mobile_status_updates, which is balanced by

the number of traditional malware pages which do use this method to throw off suspicion (which would be incurred by continuously posting links). Table 5.1 covers the robustness of our features.

Our ratings system is based off an idea from the Twitter paper by Yang et. al (Twitterevasive) and clearly demarcates three situations: *low*, if the feature is largely circumstantial and can be modified consciously with minimal to no effort cost, *medium* if modification is harder to control or can be controlled with mentionworthy effort cost, *high* if changing the feature to avoid suspicion is either out of the malware author's hands or comes at a very high price to them in terms of time or economically.

In table 5.2, we show the top 10 characteristics ranked by gain ratio, and in table 5.3 they are ranked by information gain.

In this vein, several points must be made. Generally, closeness centrality measuring only the distance to all nodes would easily be remedied on twitter since there is no limit on the number of users that another user can follow - however, the number of pages that a page can like is restricted to 25 on Facebook, so choosing between pages comes with an inherent opportunity cost. Hence we scale it up to medium. While post type and common story type are not particularly robust as far as posting methodology goes, we consider them medium-robustness because changing the posting method can potentially make it more difficult to

Table 5.1: Feature Robustness Summary

Number	Feature	Robustness
1	Verified Or Not	Medium
2	Average Ascii	Low
3	Maximum Post Similarity	Low
4	Average Post Similarity	Low
5	Average Number of Smileys	Low
6	Average Number of Non Ascii Unicode Character	Low
7	Mean Time Between Posts	Low
8	Standard Deviation of Time Between Posts	Low
9	Fan Loyalty	High
10	Page Name Length	Medium
11	Average Non-Small Letters per Post	Low
12	Average Number of Capital Letters per Post	Low
13	Average Number of Links Per Post	Medium
14	Total Number of Links	Medium
15	Number of Commonly Referenced Domains	Medium
16	Average Number Of Whitespace Chars per post	Low
17	Most Common Story	Medium
18	Most Common Post Type	Medium
19	Number of Pages Liked	Low
20	Number of Other Pages that Liked/Shared/Commented	High
21	Eccentricity	High
22	Closeness Centrality	Medium
23	Harmonic Closeness Centrality	Medium
24	Betweenness Centrality	High
25	Clustering Coefficient	Medium
26	Eigencentrality	Medium
27	Reciprocal Indegree	High
28	Reciprocal Outdegree	High
29	Reciprocal Degree	High
30	Indegree	Medium
31	Outdegree	Low
32	Degree	Medium
33	People Talking About This	Medium
34	Likes	Low
35	New Likes This Week	Low

Table 5.2: Ranking Features By Gain Ratio

Attribute Rank	Attribute	Gain Ratio
1	Indegree	0.1687
2	Degree	0.1242
3	Betweenness Centrality	0.1178
4	Reciprocal Degree	0.0998
5	Reciprocal Indegree	0.0998
6	Eccentricity	0.0998
7	Reciprocal Outdegree	0.0998
8	Closeness Centrality	0.0845
9	Harmonic Closeness Centrality	0.0845
10	Outdegree	0.0834

Table 5.3: Ranking Features By Information Gain

Attribute Rank	Attribute	Information Gain
1	Degree	0.2378
2	Indegree	0.1505
3	Outdegree	0.1117
4	Avg Post Links	0.1047
5	Total Post Links	0.094
6	Reciprocal Outdegree	0.0376
7	Reciprocal Indegree	0.0376
8	Eccentricity	0.0376
9	Reciprocal Degree	0.0376
10	Harmonic Closeness Centrality	0.0237

propagate links. Clustering coefficient would normally be considered very high in terms of robustness as outside of controlling an entire set of interconnected pages, it is nigh-impossible to set up a situation where a page is connected to a set of interconnected pages that are not sparsely linked. We drop its robustness to medium considering that a non-zero value is almost never encountered in our data set.

As part of our comparative work, we contrast the method used by Dewan et. al with

identifying malicious pages from their posts using n-gram bag of words models. Similar to theirs we attempt a similar classification and our results are in Table 5.4 compared to the n-gram classification on our dataset.

Table 5.4: Comparative Results with *Hiding in Plain Sight* [2]

	n-gram Results	k-fold cross validation (MAFIA Feature Set)
Naive Bayes	0.56	0.66
Logistic Regression	0.51	0.688
Neural Network	0.526	0.688
Random Forest	0.60	0.762

We also use Sparsenn and natural language toolkits and as we can see with a change in training dataset to reflect traditional malware, it is much less effective to use n-grams (n=1,2,3) to classify their posts.

We suspect this to be because unlike some of the other categories used by Dewan et al. as undesirable pages on Facebook, several are based off the idea of engaging the users in the form of textual posts or purely through pictures. In the absence of pages based on displaying pornographic images or politically or racially motivated arguments or groups, there is much less of a need for long text-based posts to make a coherent argument in any form. Since most malicious pages are posting urls (which break sentence structure) or promoting earlier links, this method seems to lose its efficacy. As we can see, even the most effective neural networks methods, which provided 84% in earlier works, drop down to 50%.

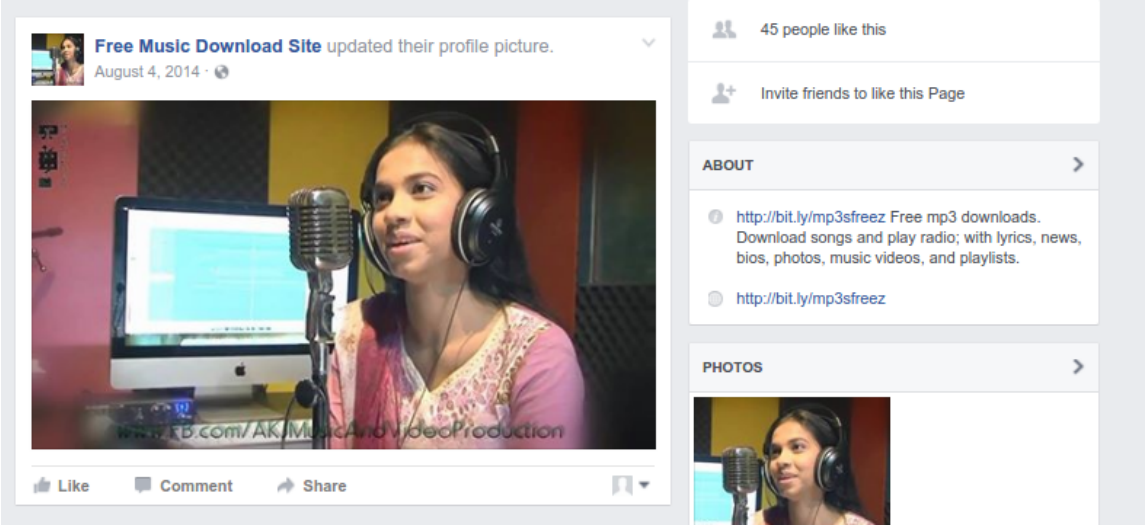


Figure 5.3: Case Study 1 - *Free Music Download Site*

5.2 Case Studies

5.2.1 Case Study 1

In this example, we compare 2 pages that are a textbook example of a mini-campaign. These two pages have similar long names ('Free Music Download Site' and 'Free Music Download Program'), and post the same link as their reference URL (http://bit.ly/mp3sfreez). They are shown in Figure 5.3 and Figure 5.4.

They also post exactly once on the same date (August 4, 2014) and have exactly 45 followers each. The page they link to offers a malicious download as confirmed by Virustotal.

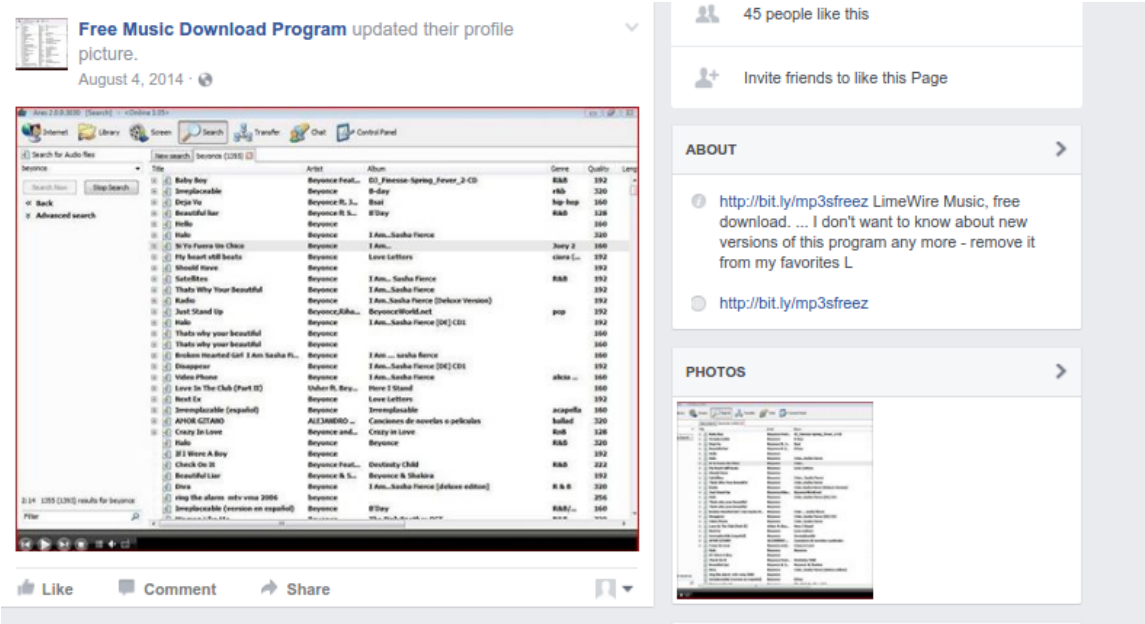


Figure 5.4: Case Study 1 - *Free Music Download Program*

5.2.2 Case Study 2

This situation involves 3 pages connected in a hub manner with two pages following the third but having no other links outward or inward. The pages have very similar names ('WhatsApp Hack - Account Spy Software' and 'Snapchat Hack - Account Spy Software'), and a combined audience of 3000 (for exactly 3 posts made on the same day, within hours of each other). These are shown in figure 5.5, and figure 5.6. Interestingly, the software offered is also remarkably similar in both cases suggesting that these are indeed made by the same individual.

Both applications are seen to be malicious actors that install a Trojan on the user's machine (shown in figure 5.7 and 5.8). They also link to a third page which offers a 'Cell

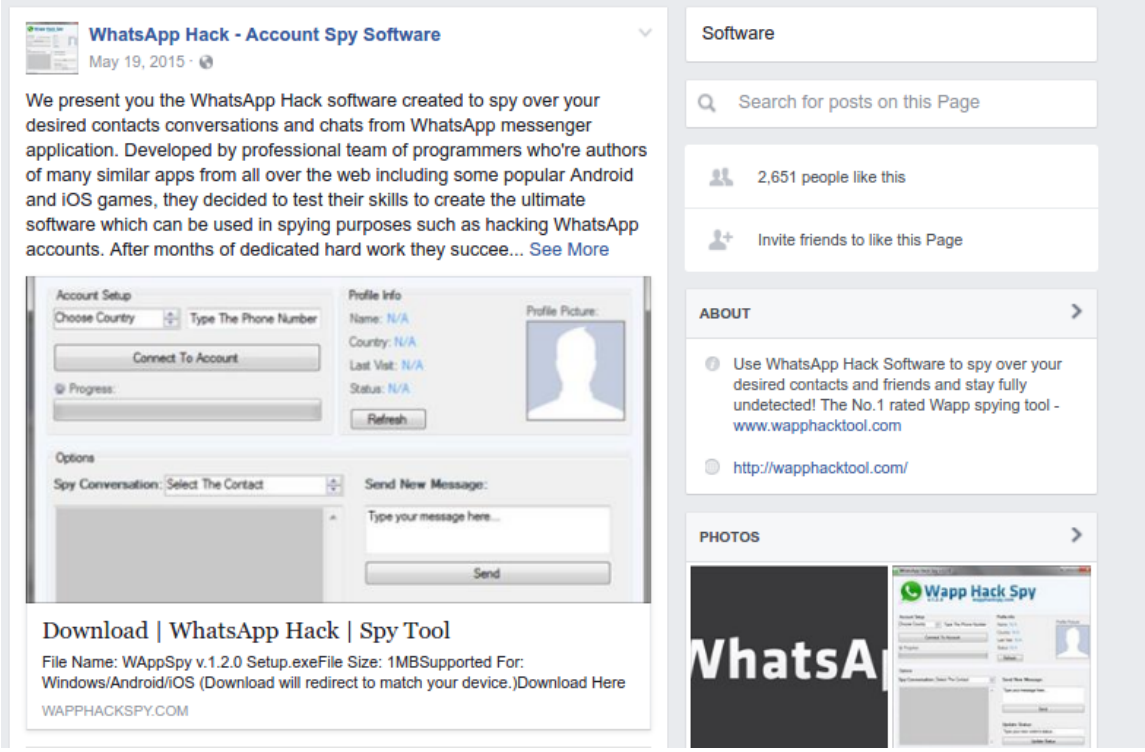


Figure 5.5: Case Study 2 - *WhatsApp Hack - Account Spy Software*

Phone Tracker' application. On further pursuing this link, the user is drawn into typical spam-socware that requests personal information and mini-survey payments before providing the offered download. These pages are presented in figure 5.9 and figure 5.10.

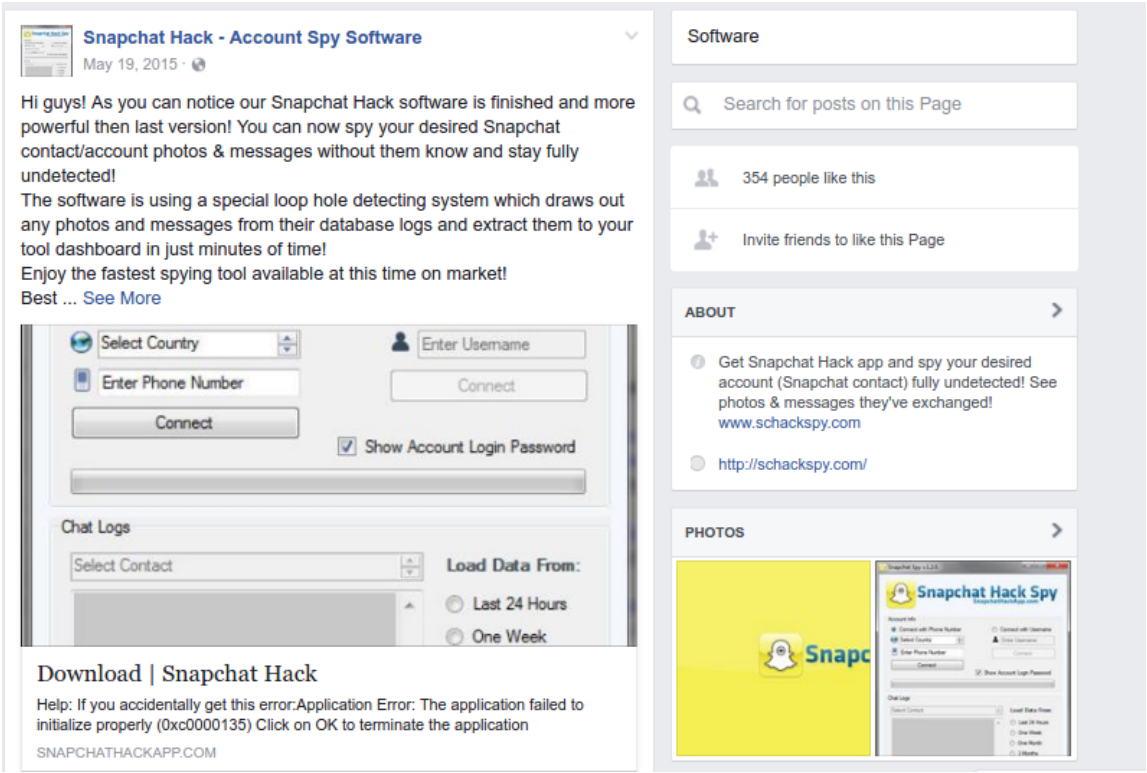


Figure 5.6: Case Study 2 - *Snapchat Hack - Account Spy Software*

Figure 5.7: Case Study 2 - *Whatsapp Hack Software Image*

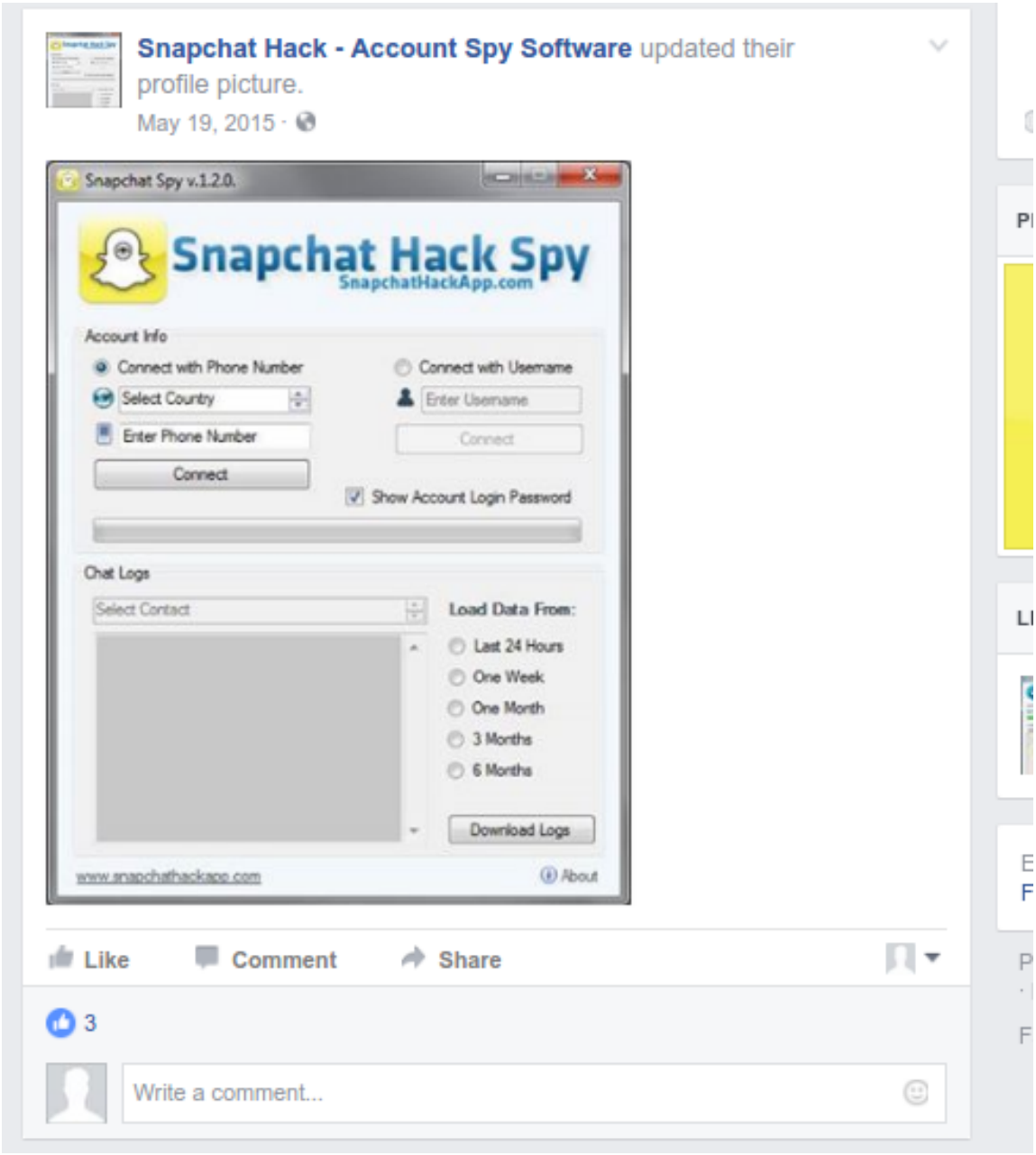


Figure 5.8: Case Study 1 - *Snapchat Hack Software Image*

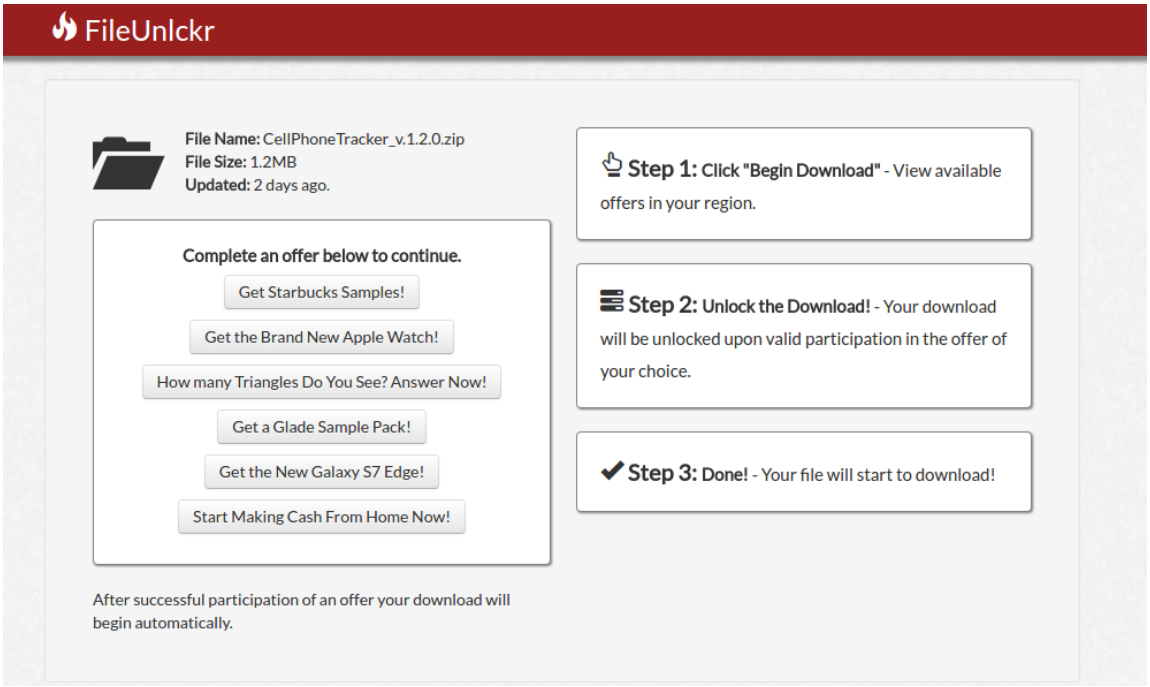


Figure 5.9: Case Study 2 - *CellPhoneTracker Download Page*

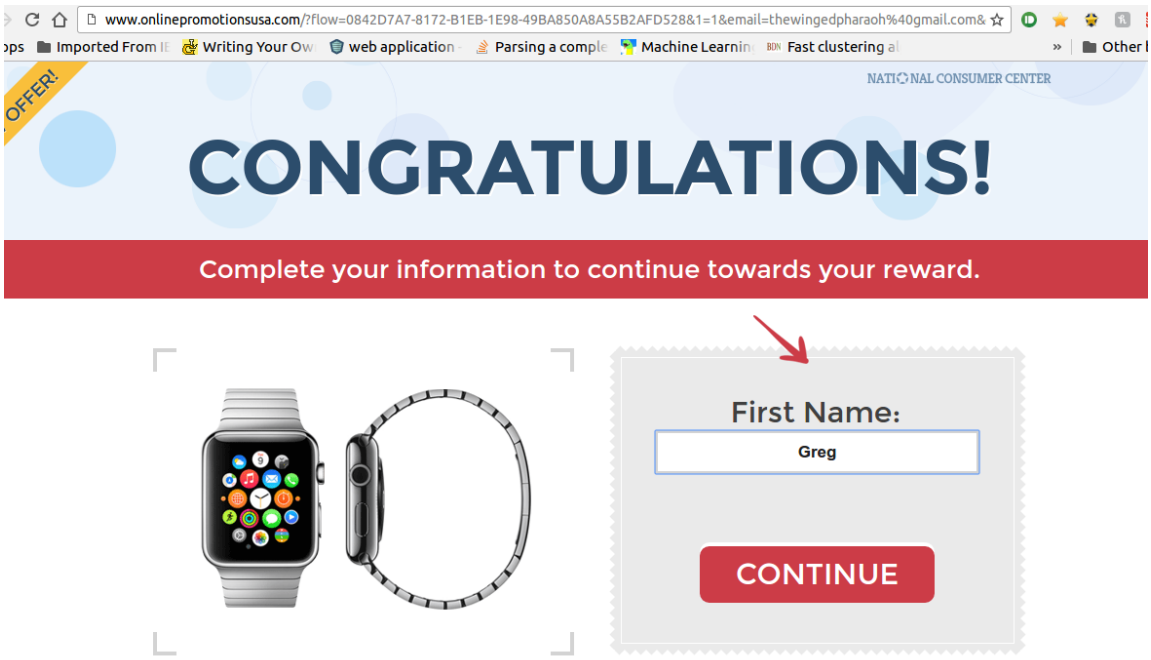


Figure 5.10: Case Study 2 - *CellPhoneTracker Socware*

6 CONCLUSIONS AND FUTURE WORK

Our work critically analyzes the Facebook ecosystem, definitely establishing the importance of pages as a key player in the malware distribution chain. We develop a resilient feature set that gives us a 81% accuracy rate with 87% AUC on the ROC graph.

We also study the nature of intra-page relationships on Facebook and discover a marked denseness in malicious page networks as compared to benign page networks which are very sparse.

Most of these pages have over 5000 likes, and several have over a million likes each - this is a combined audience of at least 15 million people (averaging over the number of malicious pages we find and accounting for overlap). The reach of these pages is what makes them a significant threat - once a person has vetted a page, since they do it of their own volition, they are unlikely to re-evaluate their decision. We examined a set of 22000 pages and our analysis presented us with 650 malicious pages - it would be a simple matter to uncover more given that we do not account for groups or communities of people on facebook which promote these pages, which is a similar problem which is not in the scope of this work.

We realise that our work could be more expansive, since we examine a small number pages (comparative to ecosystem size) as our dataset. However, several of our processes become computationally (spatially and temporally) infeasible as the dataset increases.

We could also combine our work with those of several previous studies and study

the impact of common users who liked malicious and benign pages, what the friendship relations are between them, and whether malicious pages or benign pages use more like harvesting.

REFERENCES

[1] Emiliano De Cristofaro, Arik Friedman, Guillaume Jourjon, Mohamed Ali Kaafar, and M Zubair Shafiq. Paying for likes?: Understanding facebook like fraud using honeypots. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 129–136. ACM, 2014.

[2] Prateek Dewan and Ponnurangam Kumaraguru. Hiding in plain sight: The anatomy of malicious facebook pages. *arXiv preprint arXiv:1510.05828*, 2015.

[3] Prateek Dewan and Ponnurangam Kumaraguru. Towards automatic real time identification of malicious posts on facebook. In *Privacy, Security and Trust (PST), 2015 13th Annual Conference on*, pages 85–92. IEEE, 2015.

[4] Manuel Egele, Andreas Moser, Christopher Kruegel, and Engin Kirda. Pox: Protecting users from malicious facebook applications. *Computer Communications*, 35(12):1507–1515, 2012.

[5] Hongyu Gao, Jun Hu, Christo Wilson, Zhichun Li, Yan Chen, and Ben Y Zhao. Detecting and characterizing social spam campaigns. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, pages 35–47. ACM, 2010.

[6] Chris Grier, Kurt Thomas, Vern Paxson, and Michael Zhang. @ spam: the underground on 140 characters or less. In *Proceedings of the 17th ACM Conference on Computer and Communications Security*, pages 27–37. ACM, 2010.

[7] Theus Hossmann, Franck Legendre, Paolo Carta, Per Gunningberg, and Christian Rohner. Twitter in disaster mode: Opportunistic communication and distribution of sensor data in emergencies. In *Proceedings of the 3rd Extreme Conference on Communication: The Amazon Expedition*, page 1. ACM, 2011.

[8] Md Sazzadur Rahman, Ting-Kai Huang, Harsha V Madhyastha, and Michalis Faloutsos. Efficient and scalable socware detection in online social networks. In *Presented as Part of the 21st USENIX Security Symposium (USENIX Security 12)*, pages 663–678, 2012.

[9] Md Sazzadur Rahman, Ting-Kai Huang, Harsha V Madhyastha, and Michalis Faloutsos. Frappe: detecting malicious facebook applications. In *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*, pages 313–324. ACM, 2012.

[10] Ameya Sanzgiri, Andrew Hughes, and Shambhu Upadhyaya. Analysis of malware propagation in twitter. In *2013 IEEE 32nd International Symposium on Reliable Distributed Systems*, pages 195–204. IEEE, 2013.

[11] Tao Stein, Erdong Chen, and Karan Mangla. Facebook immune system. In *Proceedings of the 4th Workshop on Social Network Systems*, page 8. ACM, 2011.

[12] Bimal Viswanath, M Ahmad Bashir, Mark Crovella, Saikat Guha, Krishna P Gummadi, Balachander Krishnamurthy, and Alan Mislove. Towards detecting anoma-

lous user behavior in online social networks. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 223–238, 2014.

[13] Chao Yang, Robert Chandler Harkreader, and Guofei Gu. Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers. In *International Workshop on Recent Advances in Intrusion Detection*, pages 318–337. Springer, 2011.

[14] Chao Yang, Jialong Zhang, and Guofei Gu. A taste of tweets: reverse engineering twitter spammers. In *Proceedings of the 30th Annual Computer Security Applications Conference*, pages 86–95. ACM, 2014.